WORKSHOP: "STOCHASTIC SIMULATIONS IN BRANCHING PROCESSES"

ERNESTO BERRÍOS-CARO

FÉLIX GEOFFROY

STOCHASTIC EVOLUTIONARY DYNAMICS | MAX PLANCK INSTITUTE FOR EVOLUTIONARY BIOLOGY

INTRODUCTION

PROGRAM OF DAY 1

- 13:00 - 15:00: **Theoretical session 1**
  - Theoretical background on stochastic processes
  - Gillespie algorithm
- 15:00 - 15:20: **Break**
- 15:20 - 17:00: **Practical session 1**
  - Implementation of the Gillespie algorithm and exercises

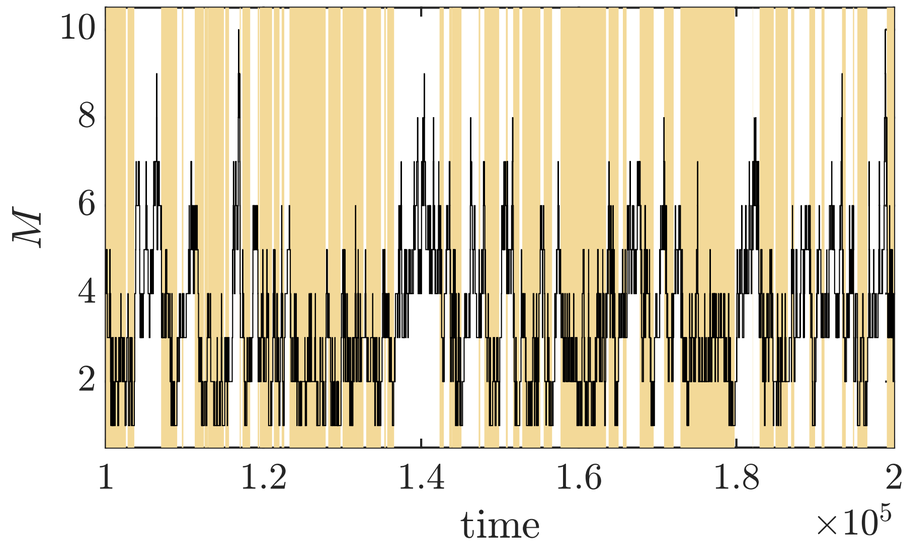
PROGRAM OF DAY 2

- 14:00 - 16:00: **Theoretical session 2**
  - Lewis' thinning algorithm
  - τ-leaping algorithm
- 16:00 - 16:20: **Break**
- 16:20 - 18:00: **Practical session 2**
  - Implementation of the τ-leaping algorithm and exercises

OVERVIEW OF THE ALGORITHMS TO BE STUDIED

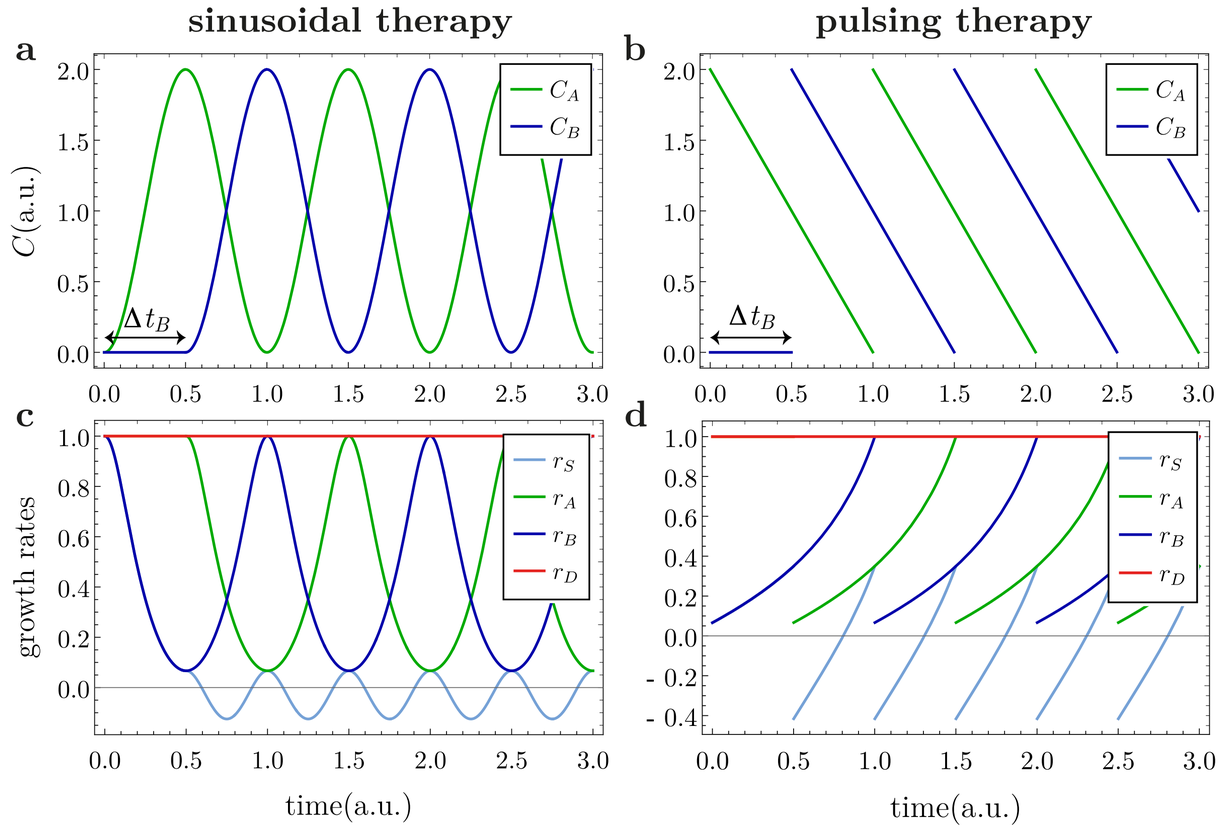
- **Gillespie algorithm** (continuous-time simulation):

  a standard method used to simulate continuous-time Markov chains with constant rates. This algorithm generates exact realisations of the system dynamics



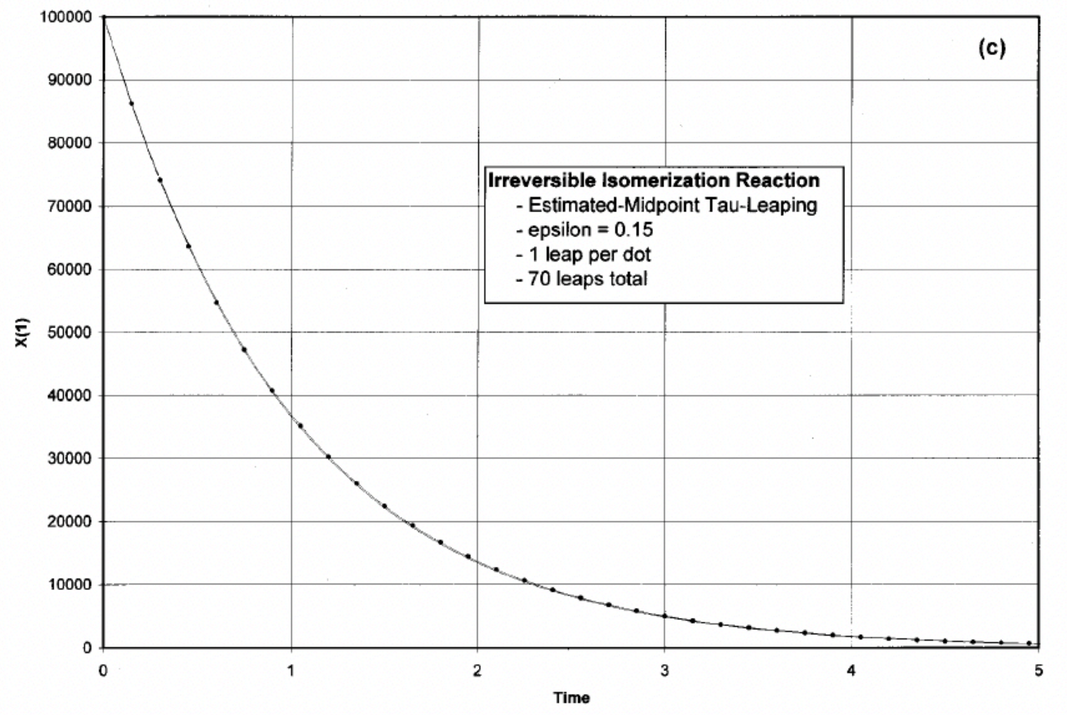
- **Lewis' thinning algorithm:**

  a modified version of the Gillespie algorithm suitable for systems with time-dependent rates.



- **τ-leaping algorithm:**

  an approximate method based on the Gillespie algorithm. This algorithm works in discrete-time and allows for a more efficient simulation in larger systems.

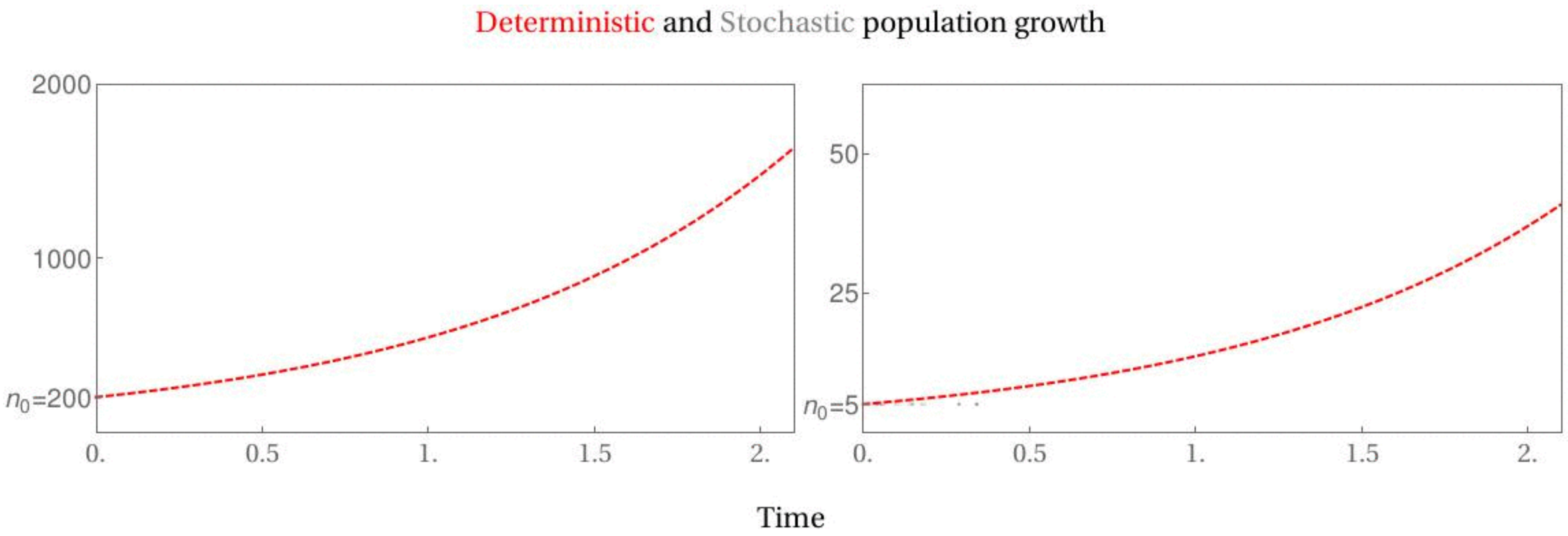


PRACTICAL SESSIONS

- A worksheet with exercises will be uploaded to the workshop event website.
- Results will be shown at the end of the session.
- We will use Python for our explanations.

THEORETICAL BACKGROUND

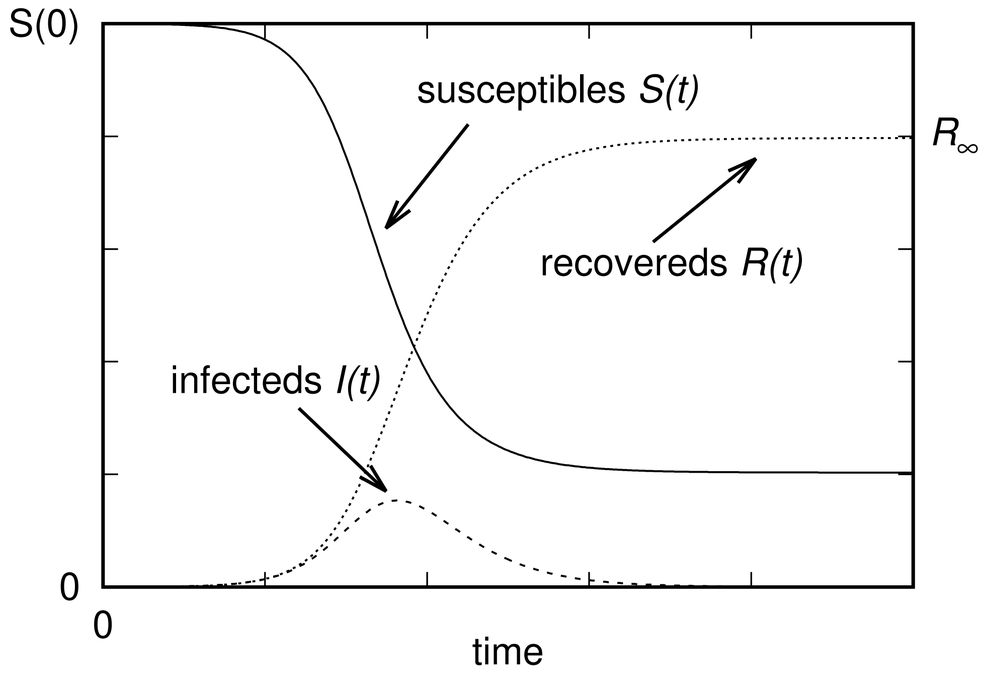
DETERMINISTIC VS STOCHASTIC PROCESSES



- **Deterministic systems**:
  - No **randomness** is involved.
  - Each realisation is the same (i.e., each run gives the same output).
  - Typically described by an ODE system.

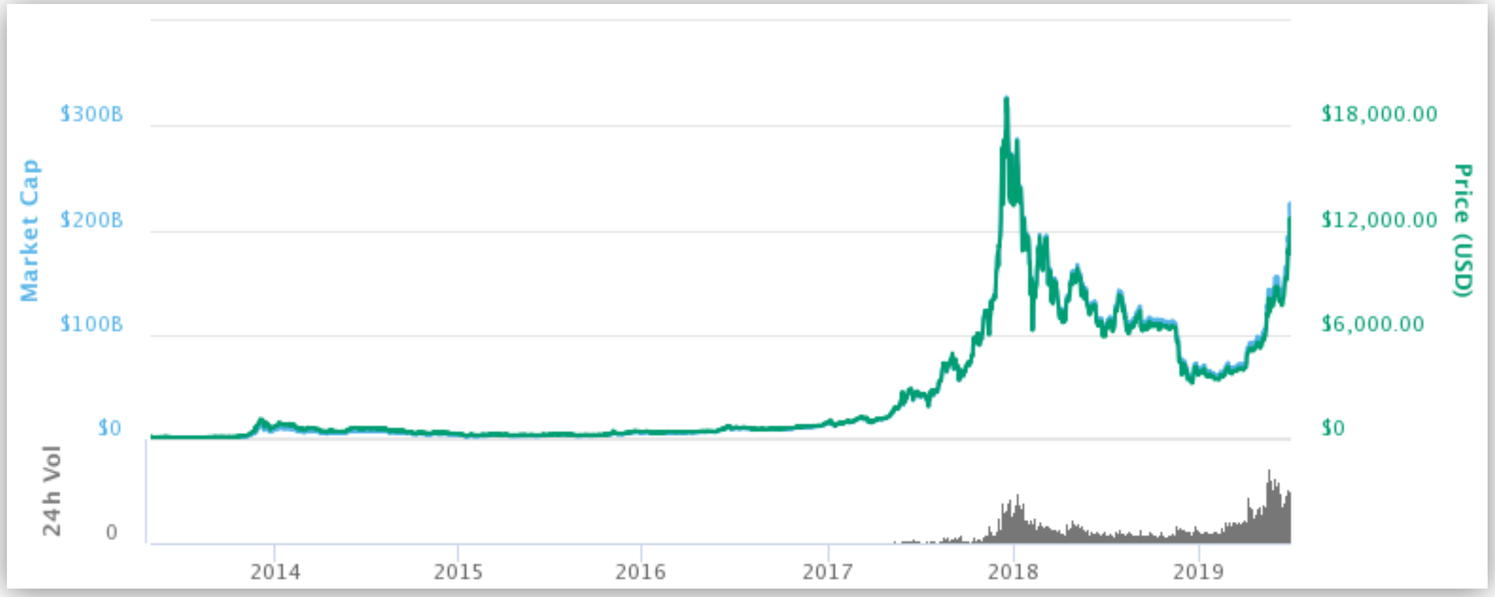
$$\frac{\mathrm{d}S}{\mathrm{d}t} = \Lambda - \beta SI - \delta_S S$$
$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta SI - \delta_I I - rI$$
$$\frac{\mathrm{d}R}{\mathrm{d}t} = rI - \delta_R R$$

deterministic ODE system



- **Stochastic systems**:
  - **Randomness** cannot be ignored
  - Each realisation is **different** (i.e., each run gives a different output).
  - Their description depends on the **type of stochasticity**.

Brownian motion



Bitcoin evolution

- Randomness can be present **extrinsically** (i.e., given by external factors).
  - Example: environmental fluctuations such as variations in temperature, pressure, drug concentrations, etc.

Ornstein-Uhlenbeck process

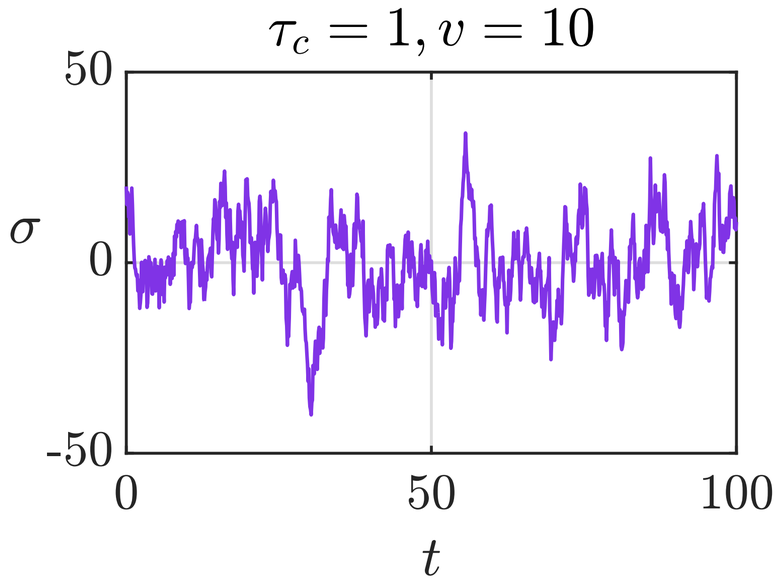
$$\frac{\mathrm{d}\sigma}{\mathrm{d}t} = \theta(m - \sigma) + s\,\eta(t)$$

deterministic approach

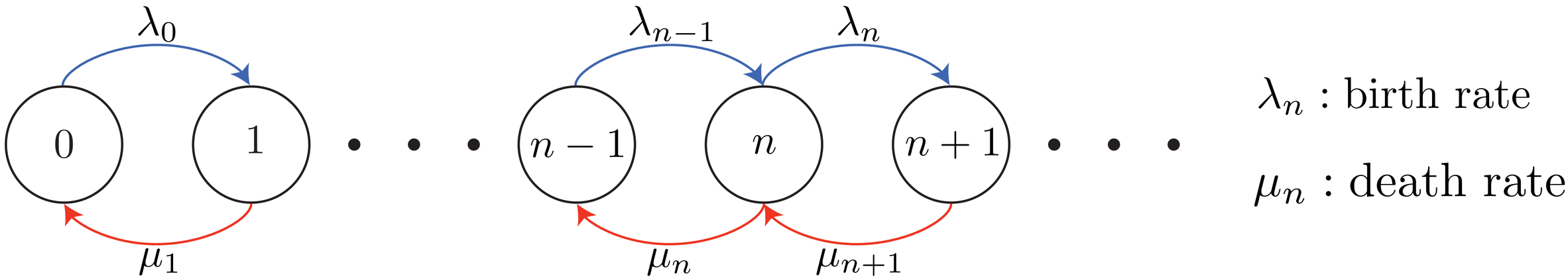
noise

Can be simulated using the

Euler-Maruyama algorithm



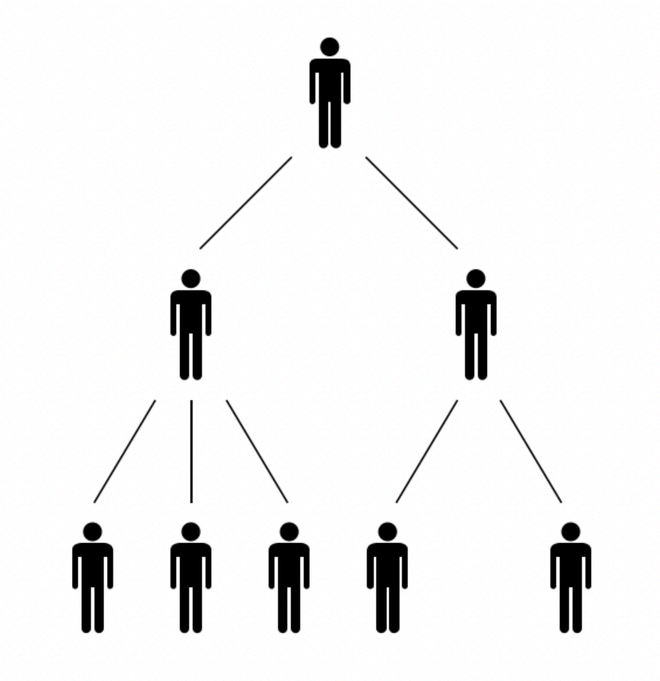
- Randomness can be present **intrinsically**, i.e., randomness is an **inherent** property of the system.
  - Example: mutations, cell duplication, cell death, etc.



we will focus on a particular case:
branching processes

BRANCHING PROCESSES

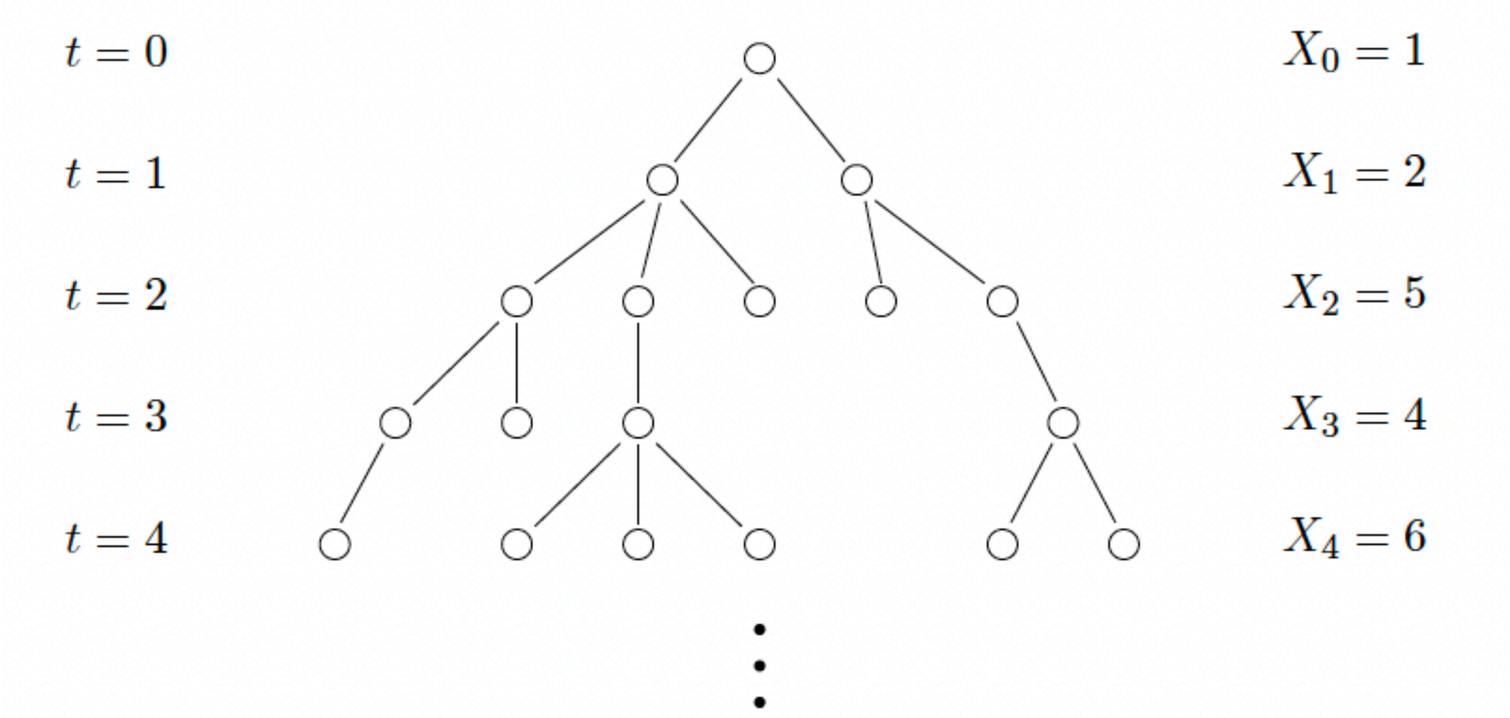
- Stochastic processes describing the reproductive dynamics of a population at the level of **individual transitions**



Each individual produces a **random number of offspring**

DISCRETE-TIME: GALTON-WATSON PROCESS

time is measured in **generations**



X_t represents the number of individuals in generation t

$Z_{i,t}$: number of offspring of individual i in generation t

offspring distribution: $p_n = \mathbb{P}(Z_{i,t} = n), \quad n = 0, 1, 2, \ldots$

- **Three properties** define a Galton-Watson process:

  ✳ All individuals are of a **single type** with identical offspring distribution.

  ✳ Individuals reproduce **independently** of each other.

  ✳ The **offspring distribution** is the same in every generation.

- Numerical simulation of Galton-Watson processes:

1. Initialise the system with $X_t = x_0$ at $t = 0$.

2. **Calculate offspring**: Draw an independent random number for each individual i present in generation t, i.e.,

$$r_i \quad \text{for} \quad i = 1, 2, \ldots, X_t$$

3. **Next generation**: Sum these random numbers to calculate the number of individuals in the next generation, i.e., $t + 1$:

$$X_{t+1} = \sum_{i=1}^{X_t} r_i$$

4. Update time from t to $t + 1$.

5. Go to 2.

```python
nb_sim = 100
nb_gen = 5
R0 = 1.5

inf_hist = np.zeros((nb_sim, nb_gen))

for i in range(nb_sim):
    nb_inf = 1
    inf_hist[i, 0] = nb_inf
    for j in range(1, nb_gen):
        nb_secondary_cases = np.random.poisson(R0, nb_inf)
        nb_inf = np.sum(nb_secondary_cases)
        inf_hist[i, j] = nb_inf

mean_inf = np.mean(inf_hist, axis=0)
for i in range(nb_sim):
    plt.plot(range(nb_gen), inf_hist[i, :], alpha=0.2)
plt.plot(range(nb_gen), mean_inf, "r--")
plt.show()
```

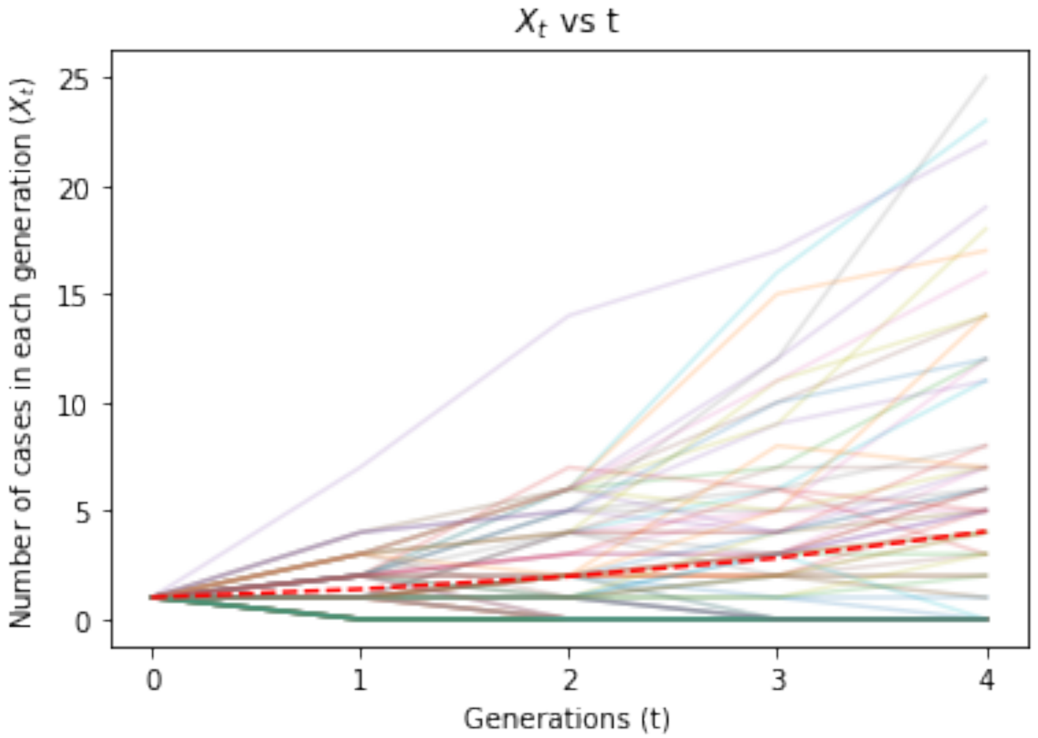
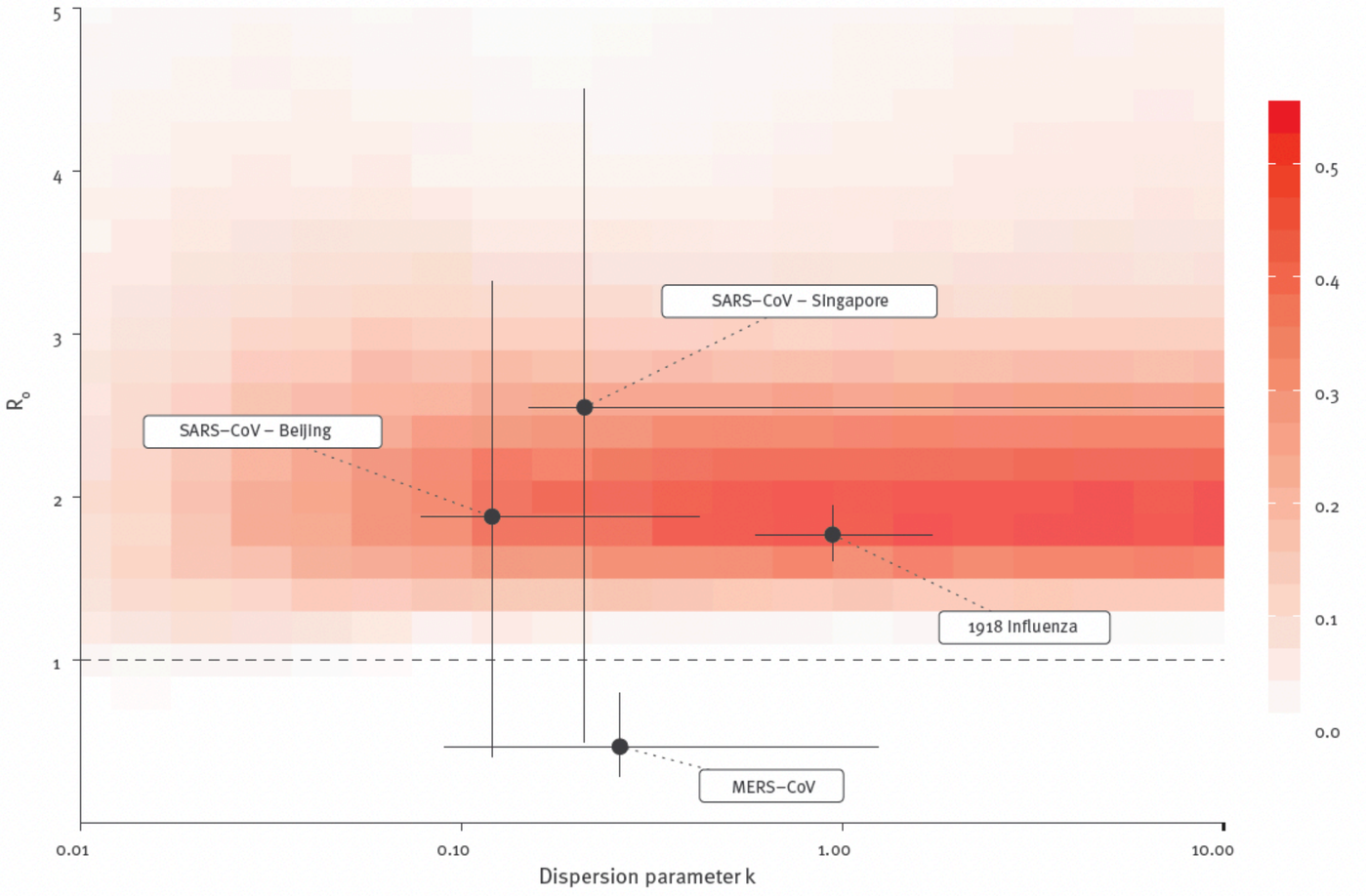


FIGURE 3

Proportion of simulated epidemics that lead to a cumulative incidence between 1,000 and 9,700 of the 2019 novel coronavirus outbreak, China, on 18 January 2020

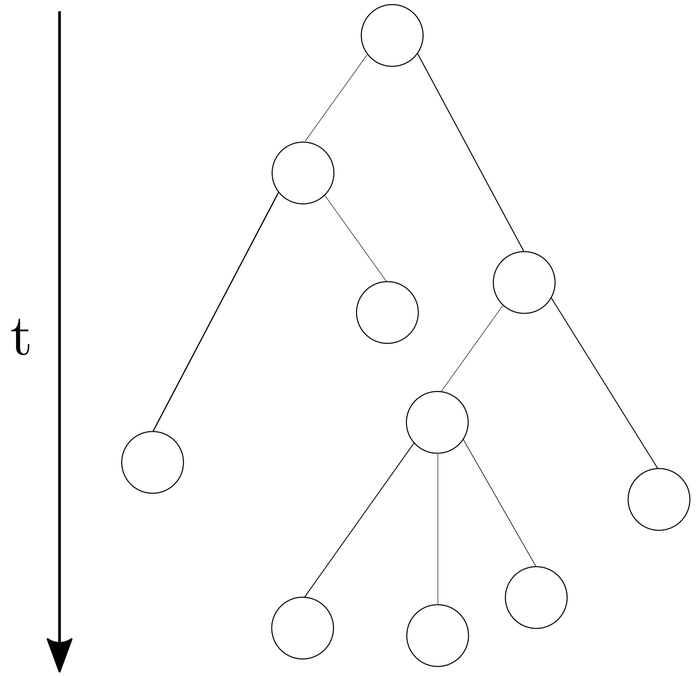


MERS: Middle East respiratory syndrome-related coronavirus; SARS: severe acute respiratory syndrome-related coronavirus.

This can be interpreted as the combinations of R_0 and k values most compatible with the estimation of epidemic size before quarantine measures were put in place. As a comparison, we show the estimates of R_0 and k for the early human-to-human transmission of SARS-CoV in Singapore and Beijing and of 1918 pandemic influenza [7,9,14].

CONTINUOUS-TIME

- Many phenomena happen in **continuous time**.



- Events **do not** occur at fixed time intervals.

- Simulations need to estimate the **time of the next event**, and **which event** will trigger.

Exact Stochastic Simulation of Coupled Chemical Reactions

Daniel T. Gillespie•

Research Department, Naval Weapons Center, China Lake, California 93555 *(Received May 12, 1977)*

Publication costs assisted by the Naval Weapons Center

JOURNAL OF CHEMICAL PHYSICS VOLUME 115, NUMBER 4 22 JULY 2001

Approximate accelerated stochastic simulation of chemically reacting systems

Daniel T. Gillespie[a)]

Research Department, Code 4T4100D, Naval Air Warfare Center, China Lake, California 93555

(Received 29 December 2000; accepted 19 April 2001)

Stochastic Simulation of Chemical Kinetics

Daniel T. Gillespie

Dan T Gillespie Consulting, Castaic, California 91384;
email: GillespieDT@mailaps.org

Annu. Rev. Phys. Chem. 2007. 58:35–55

First published online as a Review in Advance on October 12, 2006

The *Annual Review of Physical Chemistry* is online at http://physchem.annualreviews.org

This article's doi:
10.1146/annurev.physchem.58.032806.104637

0066-426X/07/0505-0035$20.00

Key Words

tau-leaping, master equation, Langevin equation, stiff systems

Abstract

Stochastic chemical kinetics describes the time evolution of a well-stirred chemically reacting system in a way that takes into account the fact that molecules come in whole numbers and exhibit some degree of randomness in their dynamical behavior. Researchers are increasingly using this approach to chemical kinetics in the analysis of cellular systems in biology, where the small molecular populations of only a few reactant species can lead to deviations from the predictions of the deterministic differential equations of classical chemical kinetics. After reviewing the supporting theory of stochastic chemical kinetics, I discuss some recent advances in methods for using that theory to make numerical simulations. These include improvements to the exact stochastic simulation algorithm (SSA) and the approximate explicit tau-leaping procedure, as well as the development of two approximate strategies for simulating systems that are dynamically stiff: implicit tau-leaping and the slow-scale SSA.

- It is common to express the different events involved as **chemical reactions**:

$$X \xrightarrow{\gamma_i} Y$$

γ_i is the per-capita (per-cell, per-individual) rate of the reaction

- The **total rate of the reaction** (or **propensity**) is the sum of γ_i over the population size of X

$$a_i = \gamma_i \cdot n_X$$

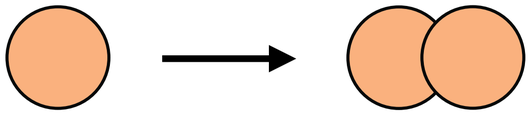
a_i is the **probability per unit time** that **any** cell of type X undertakes the reaction

- The probability that the reaction occurs during the next Δt units of time is:

$$P = a_i \Delta t$$

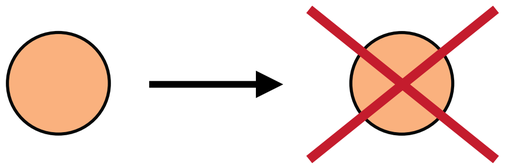
TYPICAL EXAMPLES OF REACTIONS

- Birth events



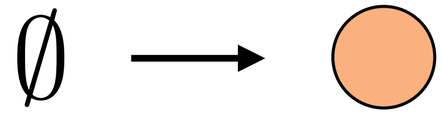
$$X \xrightarrow{b} X + X \qquad a_b = b \cdot n_X$$

- Death events



$$X \xrightarrow{d} \emptyset \qquad a_d = d \cdot n_X$$

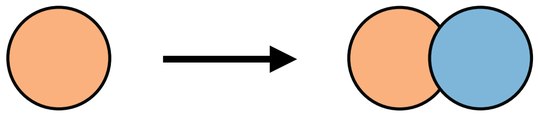
- Immigration



$$\emptyset \xrightarrow{\nu} X \qquad a_\nu = \nu$$

- Mutations

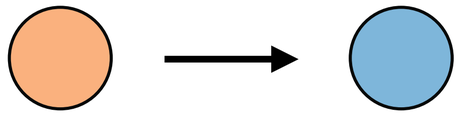
mutation through the offspring



$$X \xrightarrow{b \cdot \mu} X + Y$$

$$a_{\text{mut}} = n_X \cdot b \cdot \mu$$

direct mutation



$$X \xrightarrow{\mu} Y$$

$$a_{\text{mut}} = n_X \cdot \mu$$

- Switching environments

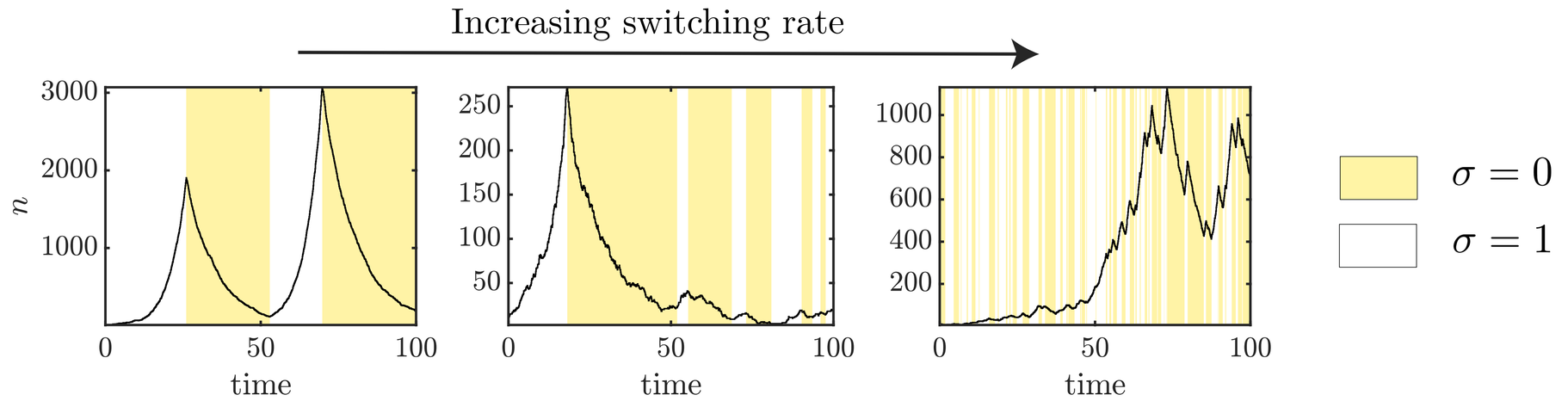
Two environments: $\sigma = 0$ $\sigma = 1$

Environmental states E_σ switch between them: $E_1 \xrightarrow{\lambda_-} E_0$ $E_0 \xrightarrow{\lambda_+} E_1$

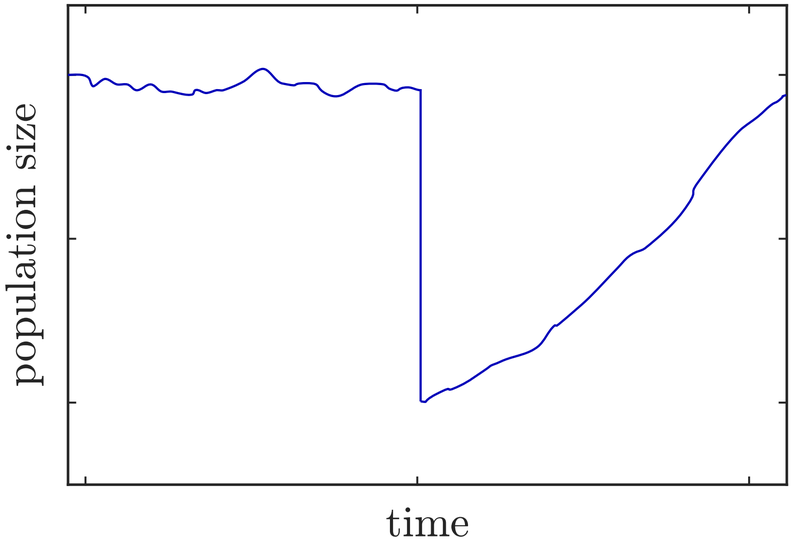
Each environment σ determines a birth rate b_σ and a death rate d_σ:

$$X \xrightarrow{b_\sigma} X + X$$

$$X \xrightarrow{d_\sigma} \emptyset$$



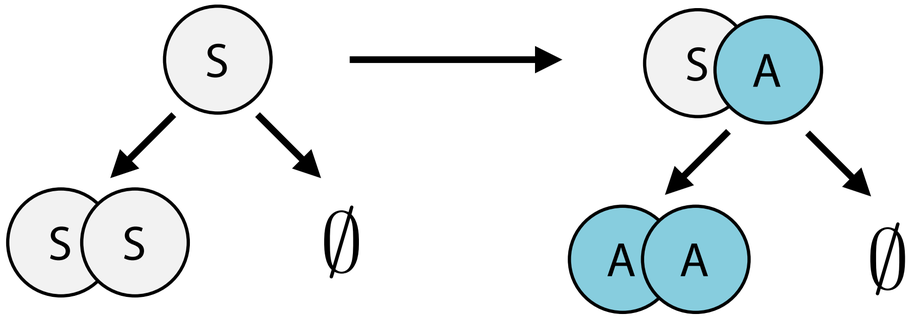
- Catastrophes



population diluted by a factor D: $n \xrightarrow{\gamma} n \cdot D$

binomially distributed bottlenecks: $n \xrightarrow{\gamma(n)} m \qquad \gamma(n) = \binom{n}{m} D^m (1-D)^{n-m}$

EXAMPLE 1: MUTATIONS



birth-death process for S:

$$S \xrightarrow{b_S \cdot (1-\mu_A)} S + S \qquad S \xrightarrow{d_S} \emptyset$$

mutation:

$$S \xrightarrow{b_S \cdot \mu_A} S + A$$

birth-death process for A:

$$A \xrightarrow{b_A} A + A \qquad A \xrightarrow{d_A} \emptyset$$

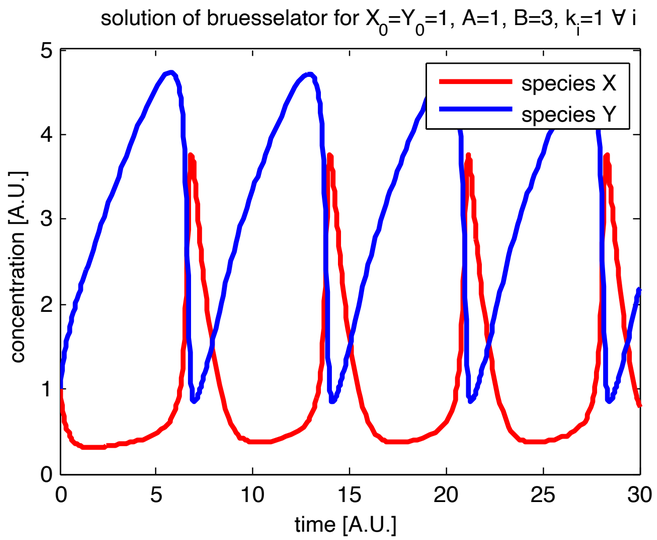
EXAMPLE 2: BRUSSELATOR

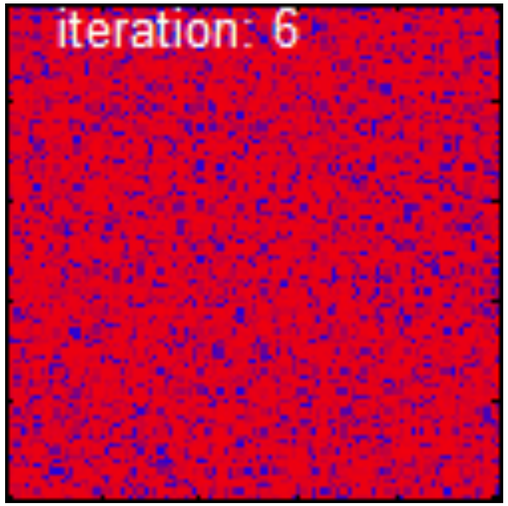
$$\emptyset \xrightarrow{1} X$$

$$X \xrightarrow{1} \emptyset$$

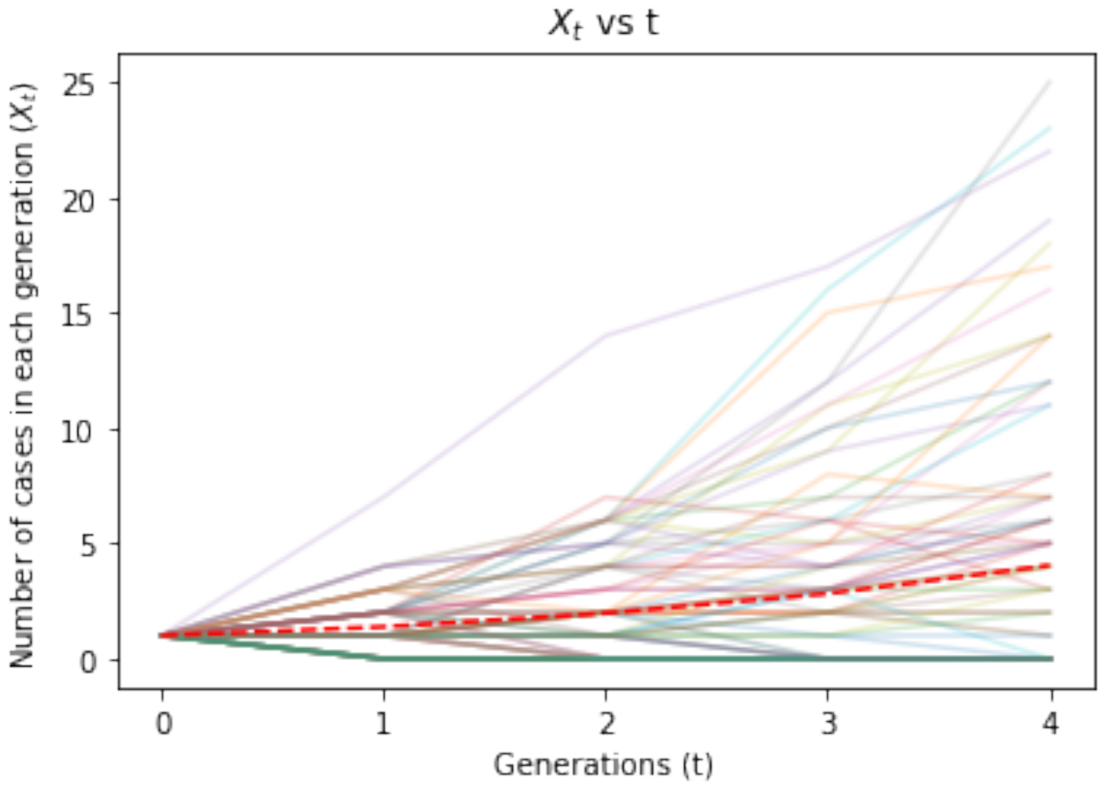
$$X \xrightarrow{b} Y$$

$$2X + Y \xrightarrow{1} 3X$$





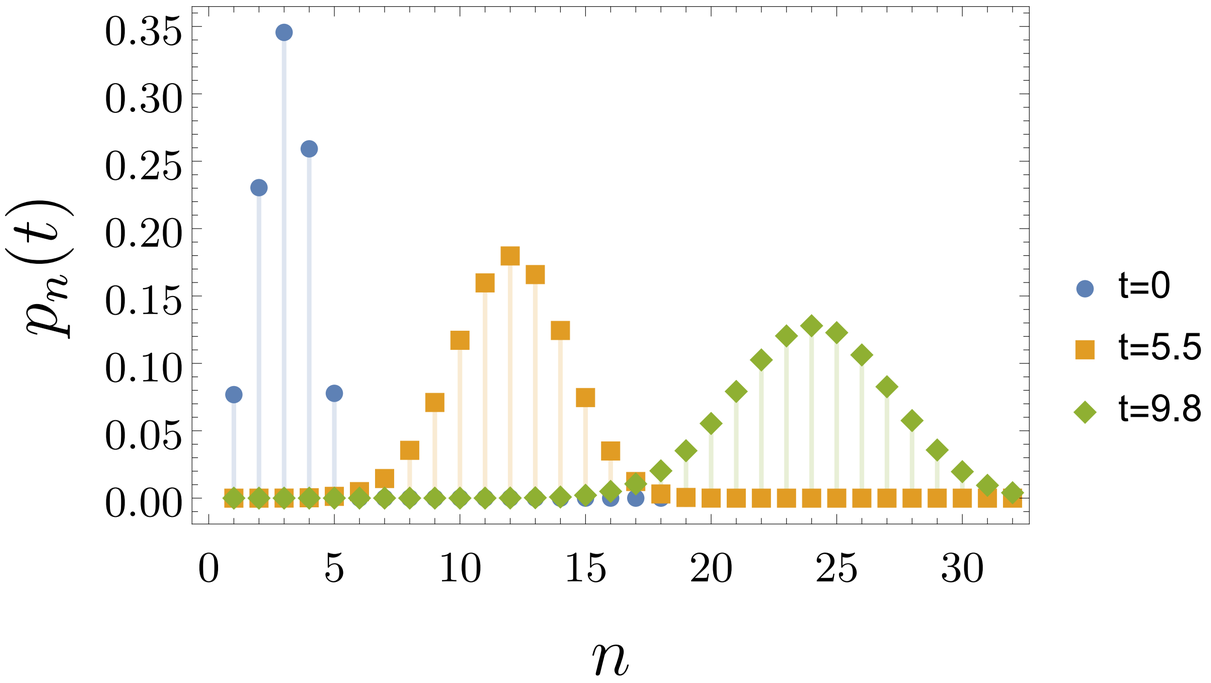
WHICH QUANTITIES CAN WE ESTIMATE FROM NUMERICAL SIMULATIONS?



RELEVANT QUANTITIES

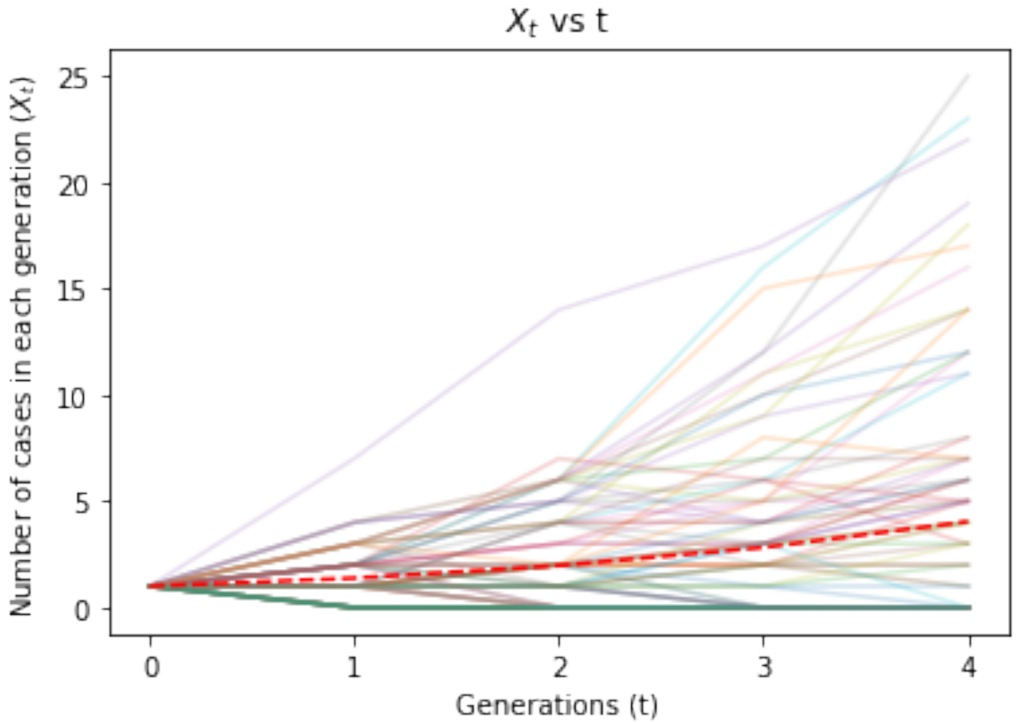
- **Probability distribution function:** probability of having n individuals at time t

$$p_n(t) = \mathbb{P}(X(t) = n)$$



- **Expected value**

$$\langle X \rangle = \sum_{n=0}^{\infty} n \cdot p_n$$



- **Variance**

$$\mathrm{Var}[X] = \langle X^2 \rangle - \langle X \rangle^2$$

- **Extinction probability:** probability of having zero individuals at time t

$$p_{\text{ext}}(t) = \mathbb{P}(X(t) = 0)$$

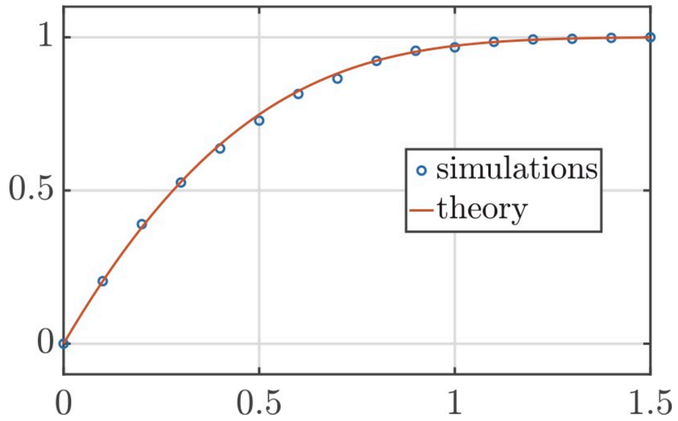
- **Ultimate extinction probability:** probability of having zero individuals as time tends to infinity

$$q = \lim_{t \to \infty} p_{\text{ext}}(t)$$

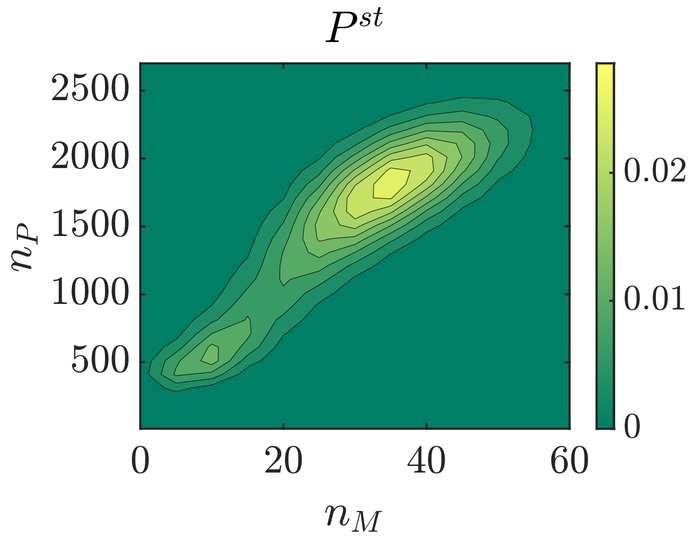
- **Expected extinction time:** expected time at which the population goes extinct

$$\langle t_{\text{ext}} \rangle$$

- **Fixation probability:** probability that a particular allele will eventually reach a frequency equal to 1

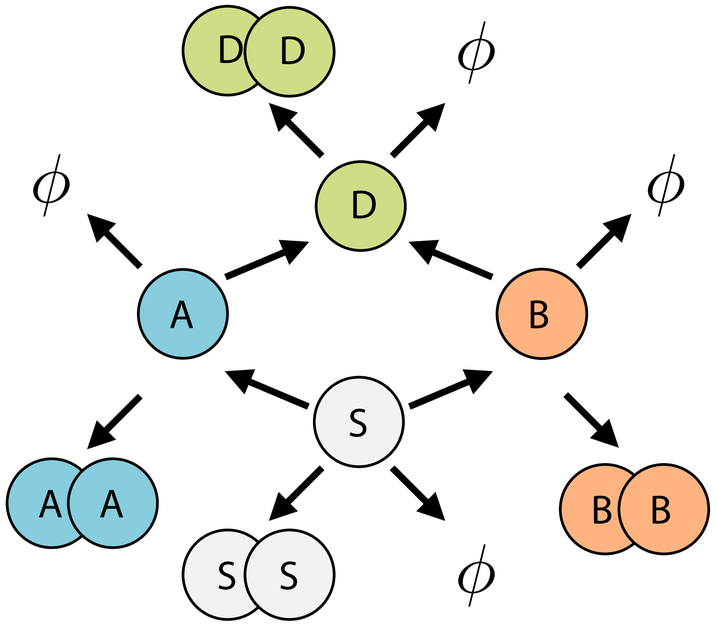


- **Sojourn time:** time the system spent in a particular space leaving it



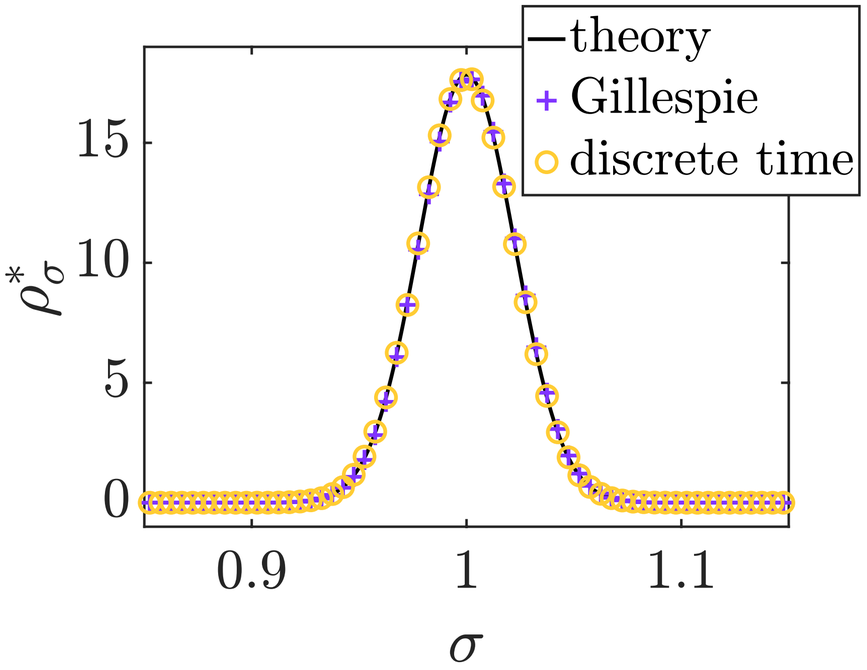
WHY DO WE NEED TO RESORT TO STOCHASTIC SIMULATIONS?

- Analytical calculations may be hard



$$
\begin{aligned}
\partial_t P(n_S, n_A, n_B, n_D, t) &= b_S(n_S - 1)(1 - \mu_A - \mu_B) P(n_S - 1, n_A, n_B, n_D, t) \\
&+ [b_S \mu_A n_S + b_A(1 - \mu_B)(n_A - 1)] P(n_S, n_A - 1, n_B, n_D, t) \\
&+ [b_S \mu_B n_S + b_B(1 - \mu_A)(n_B - 1)] P(n_S, n_A, n_B - 1, n_D, t) \\
&+ [b_A \mu_B n_A + b_B \mu_A n_B + b_D(n_D - 1)] P(n_S, n_A, n_B, n_D - 1, t) \\
&+ d_S(n_S + 1) P(n_S + 1, n_A, n_B, n_D, t) + d_A(n_A + 1) P(n_S, n_A + 1, n_B, n_D, t) \\
&+ d_B(n_B + 1) P(n_S, n_A, n_B + 1, n_D, t) + d_D(n_D + 1) P(n_S, n_A, n_B, n_D + 1, t) \\
&- [(b_S(1 - \mu_A - \mu_B) + d_S) n_S + (b_A(1 - \mu_B) + d_A) n_A \\
&+ (b_B(1 - \mu_A) + d_B) n_B + (b_D + d_D) n_D] P(n_S, n_A, n_B, n_D, t)
\end{aligned}
$$

- Numerical simulations allow us to verify our theoretical predictions



- Stochastic simulations are, in general, easy to implement

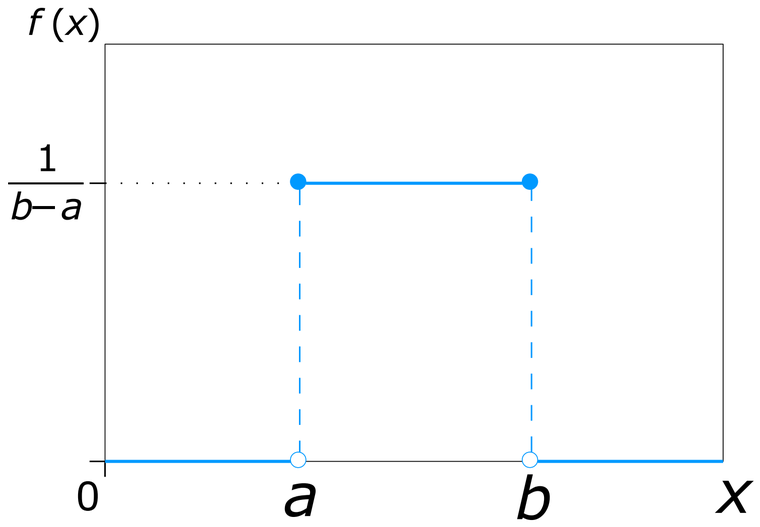
→ we can test the system dynamics before making hard calculations

RANDOM NUMBERS

```
>>> import numpy as np
```

- Relevant distributions:
  - Uniform distribution between 0 and 1

$$X \sim U(0,1)$$



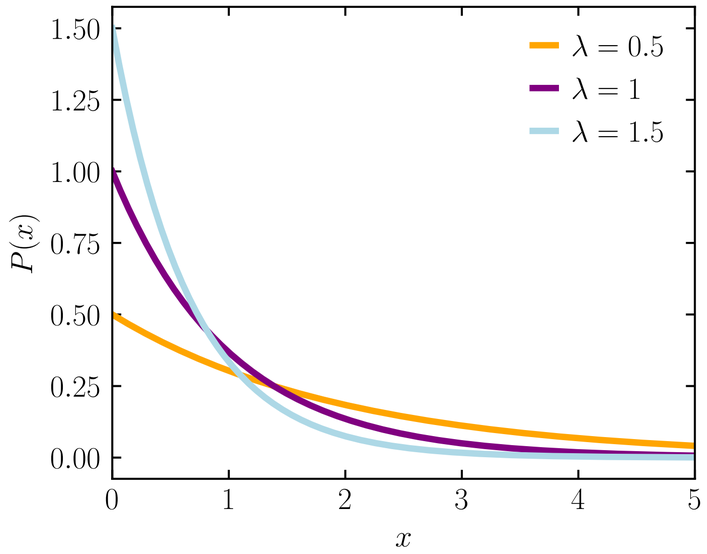

```
>>> np.random.uniform(a, b, size)
```

- Exponential distribution

$$X \sim \text{Exp}(\lambda)$$

$$p_x = \mathbb{P}(X = x) = \lambda e^{-\lambda x}$$

```
>>> np.random.exponential(1/lambda, size)
```

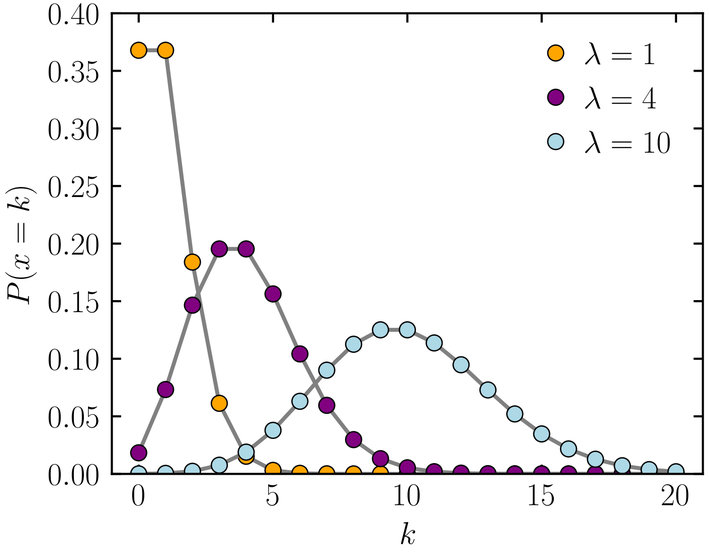


- Poisson distribution

$$X \sim \text{Poisson}(\lambda)$$

$$p_x = \mathbb{P}(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

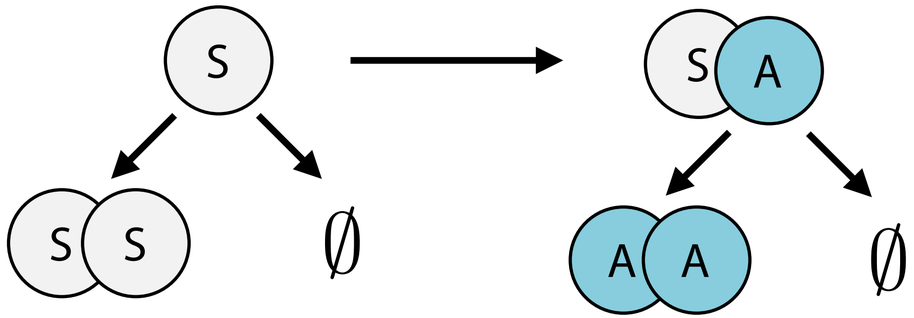
```
>>> np.random.poisson(lambda, size)
```



APPLICATIONS OF THE GILLESPIE ALGORITHM

DRUG RESISTANCE

- Consider a population with sensitive cells S and mutant cells A.
- Each strain undertakes a birth-death process. Mutations occur through the offspring.



$$S \xrightarrow{b_S \cdot (1-\mu_A)} S + S$$

$$S \xrightarrow{d_S} \emptyset$$

$$S \xrightarrow{b_S \cdot \mu_A} S + A$$

$$A \xrightarrow{b_A} A + A$$

$$A \xrightarrow{d_A} \emptyset$$

reactions	propensities
1. $S \xrightarrow{b_S \cdot (1-\mu_A)} S + S$	$a_1 = n_S \cdot b_S \cdot (1 - \mu_A)$
2. $S \xrightarrow{d_S} \emptyset$	$a_2 = n_S \cdot d_S$
3. $S \xrightarrow{b_S \cdot \mu_A} S + A$	$a_3 = n_S \cdot b_S \cdot \mu_A$
4. $A \xrightarrow{b_A} A + A$	$a_4 = n_A \cdot b_A$
5. $A \xrightarrow{d_A} \emptyset$	$a_5 = n_A \cdot d_A$

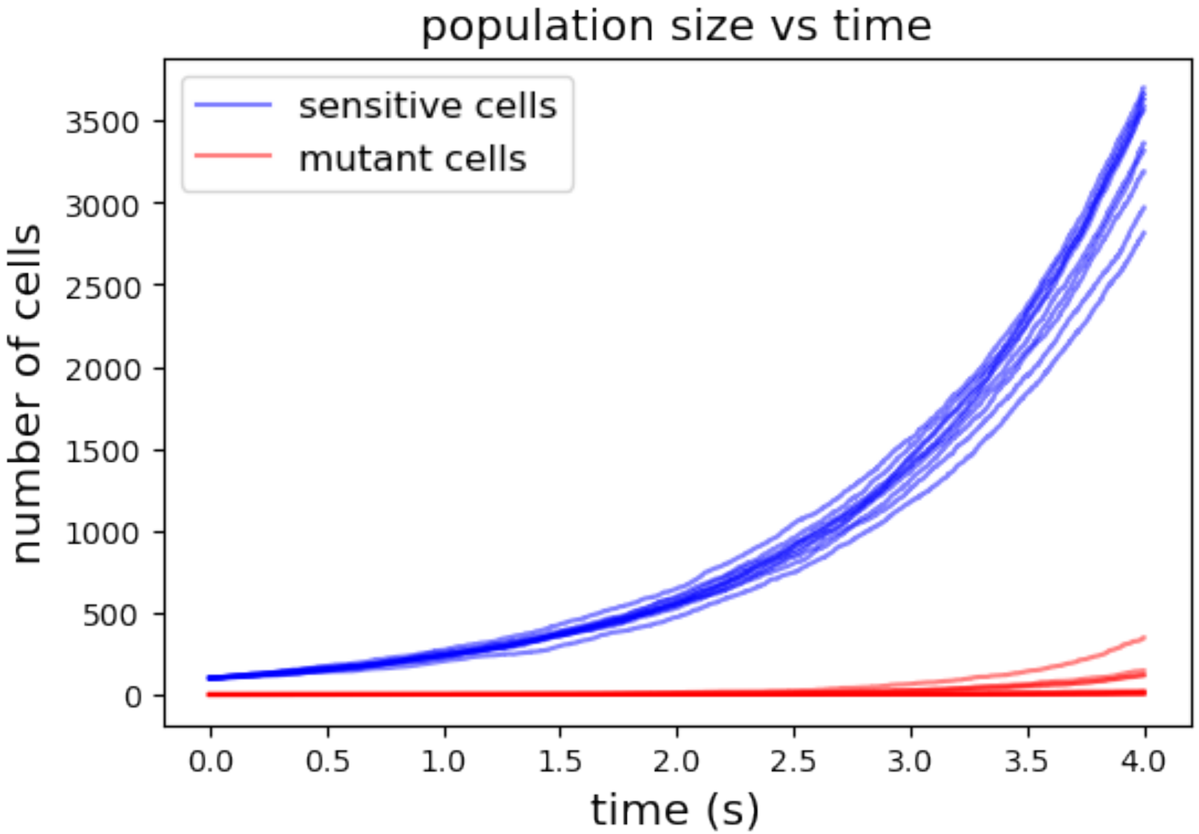
- Simulation using the Gillespie algorithm:

1. Initialise the system with $n_S = n_0$ and $n_A = 0$ at time $t = 0$.
2. **Update** every propensity a_i and the sum $a_0 = \sum_{i=1}^{5} a_i$.
3. Draw a random number $\tau \sim \text{Exp}(a_0)$. Update time to $t + \tau$.
4. Draw a random number $r \sim U(0,1)$ and execute reaction i with probability $\frac{a_i}{a_0}$.
5. Go to 2.

- How to choose the next reaction that triggers?

4. Draw a random number $r \sim U(0,1)$ and execute reaction i with probability $\frac{a_i}{a_0}$.

- If $0 \leq r < \frac{a_1}{a_0}$: execute reaction 1 $\longrightarrow$ $n_S \to n_S + 1$
- If $\frac{a_1}{a_0} \leq r < \frac{a_1 + a_2}{a_0}$: execute reaction 2 $\longrightarrow$ $n_S \to n_S - 1$
- If $\frac{a_1 + a_2}{a_0} \leq r < \frac{a_1 + a_2 + a_3}{a_0}$: execute reaction 3 $\longrightarrow$ $n_A \to n_A + 1$
- If $\frac{a_1 + a_2 + a_3}{a_0} \leq r < \frac{a_1 + a_2 + a_3 + a_4}{a_0}$: execute reaction 4 $\longrightarrow$ $n_A \to n_A + 1$
- If $\frac{a_1 + a_2 + a_3 + a_4}{a_0} \leq r \leq 1$: execute reaction 5 $\longrightarrow$ $n_A \to n_A - 1$



$$b_S = 1.0$$
$$d_S = 0.1$$
$$b_A = 1.5$$
$$d_A = 0.1$$
$$\mu_A = 10^{-3}$$
$$n_0 = 100$$

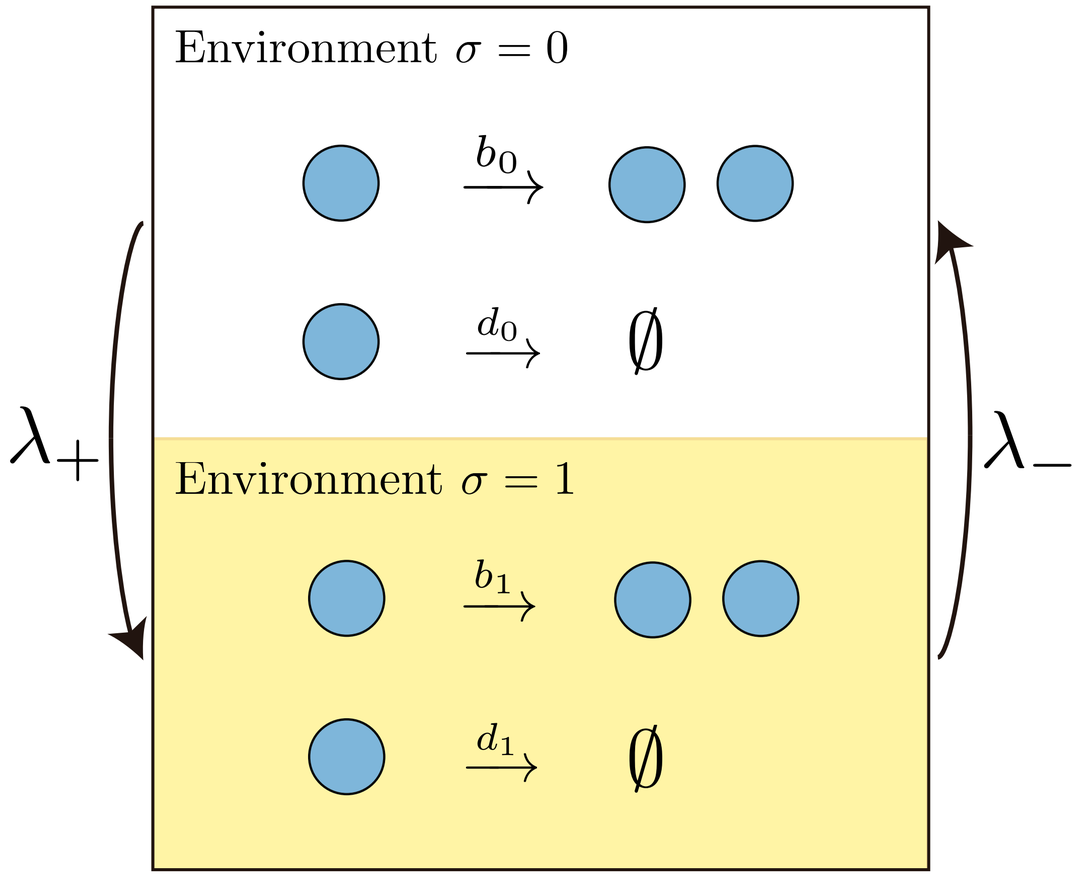
BIRTH-DEATH PROCESS + SWITCHING ENVIRONMENT

- Consider a population undertaking a birth-death process.

- Consider an external switching environment E_σ of two states $\sigma = 0$ and $\sigma = 1$. The environments switch at rates λ_+ and λ_-

$$E_1 \xrightarrow{\lambda_-} E_0 \qquad E_0 \xrightarrow{\lambda_+} E_1$$

- Birth and death rates depend on the current environment: b_σ and d_σ.



- Let us assume $b_0 > b_1$ and $d_0 = d_1$ so that $b_0 - d_0 > 0$ and $b_1 - d_1 < 0$
- How can we simulate this system using the Gillespie algorithm?

reactions	propensities
if $\sigma = 0$:	
1. $X \xrightarrow{b_0} X + X$	$a_1 = n_X \cdot b_0$
2. $X \xrightarrow{d_0} \emptyset$	$a_2 = n_X \cdot d_0$
3. $E_0 \xrightarrow{\lambda_+} E_1$	$a_3 = \lambda_+$
if $\sigma = 1$:	
4. $X \xrightarrow{b_1} X + X$	$a_4 = n_X \cdot b_1$
5. $X \xrightarrow{d_1} \emptyset$	$a_5 = n_X \cdot d_1$
6. $E_1 \xrightarrow{\lambda_-} E_0$	$a_6 = \lambda_-$

- Simulation using the Gillespie algorithm:

1. Initialise the system with $n_X = n_0$ and $\sigma = 0$ at time $t = 0$.

2. If $\sigma = 0$, **update** propensities a_i for $i = 1,2,3$ and the sum $a_0 = \sum_{i=1}^{3} a_i$.

   Draw a random number $\tau \sim \text{Exp}(a_0)$. Update time to $t + \tau$.

   Draw a random number $r \sim U(0,1)$ and execute reaction i with probability $\frac{a_i}{a_0}$ for $i = 1,2,3$.

If $\sigma = 1$, **update** propensities a_i for $i = 4,5,6$ and the sum $a_0 = \sum_{i=4}^{6} a_i$.

Draw a random number $\tau \sim \text{Exp}(a_0)$. Update time to $t + \tau$.

Draw a random number $r \sim U(0,1)$ and execute reaction i with probability $\frac{a_i}{a_0}$ for $i = 4,5,6$.

3. Go to 2.

$$b_0 = 0.3$$
$$d_0 = 0.1$$
$$b_1 = 0.1$$
$$d_1 = 0.2$$
$$n_0 = 10$$

same switching rate for both environments: $\lambda_- = \lambda_+$